ACCEPTED MANUSCRIPT

Optimizing quantum phase estimation for the simulation of Hamiltonian eigenstates

Optimizing quantum phase estimation for the simulation of Hamiltonian eigenstates

P. M. Q. Cruz[1,2,*], G. Catarina[1], R. Gautier[1,3], J. Fernández-Rossier[1,†]

[1]*QuantaLab, International Iberian Nanotechnology Laboratory (INL),*
Av. Mestre José Veiga, 4715-330 Braga, Portugal
[2]*Departamento de Física e Astronomia, Faculdade de Ciências da Universidade do Porto, 4169-007 Porto, Portugal*
[3]*CentraleSupélec, Université Paris-Saclay, 91190 Gif-Sur-Yvette, France*

(Dated: July 24, 2020)

We revisit quantum phase estimation algorithms for the purpose of obtaining the energy levels of many-body Hamiltonians and pay particular attention to the statistical analysis of their outputs. We introduce the mean phase direction of the parent distribution associated with eigenstate inputs as a new post-processing tool. By connecting it with the unknown phase, we find that if used as its direct estimator, it exceeds the accuracy of the standard majority rule using one less bit of resolution, making evident that it can also be inverted to provide unbiased estimation. Moreover, we show how to directly use this quantity to accurately find the energy levels when the initialized state is an eigenstate of the simulated propagator during the whole time evolution, which allows for shallower algorithms. We then use IBM Q hardware to carry out the digital quantum simulation of three toy models: a two-level system, a two-spin Ising model and a two-site Hubbard model at half-filling. Methodologies are provided to implement Trotterization and reduce the variability of results in noisy intermediate scale quantum computers.

I. INTRODUCTION

The computational resources required to model a quantum system in a classical computer scale exponentially with the number of degrees of freedom. This is known as the exponential wall problem [1]. As a result, complete numerical solutions of the general many-body problem, where reduction schemes for the Hilbert space of the system are impossible or unknown, can only be achieved for very small systems. This precludes the simulation of interesting molecules and their chemical reactions with the so-called chemical accuracy [2]. Finding a viable universal approach to solve the many-body problem, going around the exponential wall, would enable tremendous progress in scientific areas such as condensed matter physics and quantum chemistry.

In this context, Feynman put forward the notion that quantum simulation should be used to circumvent the exponential wall [3], even before quantum computers were envisioned [4]. Once the concept of the gate based universal quantum computer was established, quantum algorithms were proposed [5, 6] to tackle the many-body problem. This approach to the simulation of quantum systems is referred to as Digital Quantum Simulation (DQS) [7], to distinguish it from analog quantum simulators [8]. In the context of quantum chemistry, DQS enables the computation of molecular energies [9–11] or other physical quantities [12]. In general, DQS could be used to tackle the many-body problem both in condensed matter [9, 12–14] and quantum field [15] theories.

There are now several quantum and quantum-classical algorithms that would permit to address the many-body problem if sufficiently powerful quantum computers were available. Outstanding examples include the simulation of time evolution [16] and the variational quantum eigensolver (VQE) [17–19]. The latter is a heuristic approach to find the groundstate energy by classically optimizing the parameters of a quantum ansatz. This hybrid algorithm has been more recently proposed as a better suited methodology for Noisy Intermediate Scale Quantum (NISQ) [7] computing devices. However, the former strategy is still worth pursuing for being an exact method to approximate quantum dynamics which, when combined with the Phase Estimation Algorithm (PEA) [20, 21] or the Iterative PEA (IPEA) [22, 23], permits to obtain the full energy spectrum of many-body Hamiltonians [9–11], unlike the VQE.

In the last few years, quantum computing hardware has experienced a qualitative leap with the advent of cloud-based quantum computing platforms. Motivated both by the availability of quantum hardware and the potential of quantum computing to tackle the many-body problem, we explore the implementation of Quantum Phase Estimation (QPE) based DQS algorithms in the NISQ computers of IBM [24]. At the time of writing, the IBM Q platform permits remote access to quantum computers with 5, 16, 20 and 53 superconducting qubits, and is being used by dozens of research groups worldwide to explore a broad set of applications [13, 25–31].

Phase estimation procedures are very important to determine the eigenvalues of a given unitary operator [20, 21, 32]. Their applications span several areas including factorization [33], sensing [34], gate calibration [35] and, relevant for this work, quantum simulation [9–11, 36]. There are two main strategies for algorithmic QPE. The first makes use of the gate expensive inverse quantum Fourier transform (QFT) [20] and, in an ideal quantum computer, could work with a single shot read-

───────

[*]pedro.cruz@quantalab.uminho.pt
[†]On leave from Departamento de Física Aplicada, Universidad de Alicante, 03690 San Vicente del Raspeig, Spain.

out. The second uses much shallower circuits, such as the one proposed by Kitaev [32] or the so-called iterative PEA [22, 23], but requires multiple readouts and classical processing.

There is a large body of work dedicated to optimize both variants of QPE algorithms. In [37], an extension of Kitaev's approach [32] was studied, offering a logarithmic reduction in the number of necessary measurements to estimate the phase with exponential accuracy. More recently, a post-processing method based on classical time-series analysis of QPE measurements was shown numerically [36] to be capable of determining multiple eigenvalues simultaneously when initializing general quantum states. Another approach to estimate several eigenvalues simultaneously based on time-series methods is introduced in [38]. Here, we go back to the original QFT-based QPE and introduce a novel methodology to optimize its use for NISQ computers. Our approach is based on a new estimator which can be employed to find eigenvalues of both hermitian and unitary operators. We focus on the former use case and examine the determination of the energy levels of three simple model Hamiltonians: 1) a two-level system, 2) a two-spin Ising model and 3) the two-site Hubbard model at half-filling. Our results have implications in a broader context, given that QPE algorithms are a central subroutine in quantum computing [21].

The successful implementation of the PEA-based quantum simulation requires a relatively modest number of qubits, but a rather large number of quantum gates. The number of qubits is determined by the number of single-particle states, plus a small overhead to readout the results. Given that the largest exact classical computations of fermionic Hamiltonians cannot deal with more than 30 single-particle states, 50 qubits would be enough to achieve quantum supremacy in the context of DQS. However, current state-of-the-art hardware is far from the depth required to make PEA-based DQS work beyond the simple models considered in this work.

The rest of this paper is organized as follows. In section II we summarize the main steps of DQS based on QPE algorithms. In section III we review the PEA and the IPEA and present our approach for the classical post-processing of the quantum measurements that permits to improve the DQS results. In section IV we introduce two simple spin model Hamiltonians to which the DQS is carried out, as well as the gate implementation of their unitary evolution operators. In section V we present the quantum computation results obtained for the spin models. In sections VI and VII we introduce the two-site Hubbard model at half-filling and the quantum simulation results, respectively. In section VIII we wrap up the main results and conclusions. Additional technical details are provided in the appendices.

II. DQS VIA QUANTUM PHASE ESTIMATION

In this section, we summarize the theory of DQS based on QPE algorithms that permit to obtain the eigenvalues of model Hamiltonians. In particular, we illustrate the main concepts using the PEA. A prerequisite is to be able to encode the quantum states of the Hamiltonian in qubit states. In the case of spin $S = 1/2$ models, this is straightforward. In the case of fermions, there are canonical transformations such as the Jordan-Wigner transformation that provide a mapping between second quantization fermion operators and spin operators.

The PEA is schematically shown in Fig. 1. The circuit uses two registers: the *phase register*, on top, has R qubits ($R = 3$ in the case of the diagram) to encode a phase approximation with one of the 2^R possible readout states; the lower line(s) correspond to the multi-qubit *simulation register* where a relevant many-body state is loaded. The procedure has the following steps:

1. *Initialization.* An eigenstate $|\phi\rangle$ of the target Hamiltonian $\mathcal{H}$, with energy ε, is prepared in the lower register of the diagram. When the state is known, there is an algorithm to carry out this task [39]. Except for simple model Hamiltonians, eigenstates are in general unknown and therefore the input is a linear combination of eigenstates. The consequences are discussed in the next section.

2. *Unitary evolution.* This subroutine entails the representation of (powers of) the unitary evolution operator $\mathcal{U} = e^{-i\mathcal{H}\tau/\hbar}$ into gates that act on $|\phi\rangle$ as $\mathcal{U}|\phi\rangle = e^{i2\pi\phi}|\phi\rangle$ and are controlled by the top register qubits. Most often, $\mathcal{H}$ is a sum of noncommuting terms and a Trotter-Suzuki approximation is required to implement $\mathcal{U}$. It is convenient to consider dimensionless variables for energy, $\varepsilon \to \varepsilon\,\varepsilon_0$, and time, $\tau \to \tau\hbar/\varepsilon_0$, so that the dynamical argument of the propagator is given by $2\pi\phi(\tau) = -\varepsilon\tau$. Importantly, in DQS, τ is a simulated parameter rather than actual computational time.

3. *Phase estimation.* The core of the PEA has two stages. First, the controlled $\mathcal{U}^{2^{k-1}}(\tau)$ operations kick out the fractional digits [21] of the phase to the top register. Second, the inverse Fourier transform permits to obtain them through a measurement in the computational basis, so that readout gives access to the phase $\phi(\tau)$. Repeating the procedure for several values of the control time parameter τ, the eigenvalue ε is obtained from the variation of the phase with τ,

$$\varepsilon = -2\pi\frac{\Delta\phi}{\Delta\tau}. \tag{1}$$

4. *Post-processing.* Two main obstacles arise in the previous program. First, even for an ideal quantum computer, both the fact that the initial state is

in general a linear superposition of $|\phi_n\rangle$ eigenstates and the fact that ϕ_n cannot always be expressed as a binary fraction decomposition, lead to quantum dispersion of the readouts. Second, noise and readout errors have to be dealt with when using NISQ computers. Thus, the statistical post-processing of QPE algorithms becomes very important and is the subject of the next section.
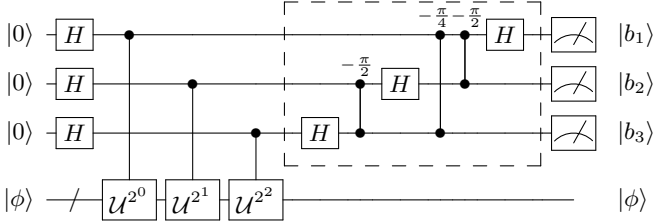

FIG. 1: Quantum circuit that implements the Phase Estimation Algorithm with 3-bit resolution. $\mathcal{U}$ is a general unitary operator that acts on a multi-qubit eigenstate $|\phi\rangle$ as $\mathcal{U}|\phi\rangle = e^{i2\pi\phi}|\phi\rangle$. In the context of quantum simulation, $\mathcal{U}$ is the time evolution propagator of the simulated Hamiltonian. The boxed operations illustrate the permuted inverse quantum Fourier transform. The output is a rational approximation to ϕ given by $0.b_1 b_2 b_3$.

III. POST-PROCESSING

This section is devoted to the statistical methods that we have implemented to enhance the phase estimation procedure described in the previous section. Before presenting the methods, we briefly review some technical aspects of QPE algorithms.

We explore two different QPE algorithms to obtain Hamiltonian eigenvalues: the so-called PEA [21] and IPEA [22, 23]. Both address the following problem: given an unitary operator $\mathcal{U}$, and one of its eigenstates $|\phi\rangle$, such that $\mathcal{U}|\phi\rangle = e^{i2\pi\phi}|\phi\rangle$, PEA and IPEA yield an estimate of the phase ϕ mod 1.

In order to approximate a real positive number smaller than 1, a binary fraction representation of ϕ with R bits is used,

$$0.\phi = b_1 2^{-1} + b_2 2^{-2} + b_3 2^{-3} + \cdots + b_R 2^{-R}. \quad (2)$$

For a given value of R, this representation defines a grid of rational numbers separated by $\delta\phi = 2^{-R}$. For instance, in the case of $R = 3$ (see circuit of Fig. 1), there are 8 possible readout states, $|000\rangle$, $|001\rangle$, $|010\rangle$, $|011\rangle$, $|100\rangle$, $|101\rangle$, $|110\rangle$, and $|111\rangle$, that represent 0, 0.125, 0.250, 0.375, 0.5, 0.625, 0.75, and 0.825, respectively.

The Hadamard gates acting on the phase register prepare all the qubits in the state $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$. The controlled $\mathcal{U}^{2^{k-1}}$ operators kick-back [20, 21] a phase into the kth qubit of the phase register $\frac{1}{\sqrt{2}}|0\rangle + e^{i2\pi 2^{k-1}\phi}|1\rangle$. The permuted inverse quantum Fourier Transform subroutine

[21] transforms this state into $|b_1\rangle \otimes |b_2\rangle \otimes \cdots \otimes |b_R\rangle$. Thus, readout of the phase register retrieves a rational approximation to ϕ.

A. Quantum dispersion of the PEA readouts

Even if we have an ideal fault-tolerant quantum computer, there are two sources of uncertainty in the implementation of the PEA. First, if the eigenvalue ϕ is not exactly equal to a binary fraction representation $0.\phi$ then different such outputs can be obtained, with probabilities given by [21]

$$P_\phi(0.\phi) = \begin{cases} 1 & , \ \phi = 0.\phi \\ \frac{1}{2^{2R}} \frac{\sin^2\left(2^R \pi(\phi - 0.\phi)\right)}{\sin^2\left(\pi(\phi - 0.\phi)\right)} & , \ \phi \neq 0.\phi \end{cases}, \quad (3)$$

where R is the number of resolution bits used. It is apparent that, as R increases, the accuracy of the procedure improves.

The second caveat in the implementation of the PEA comes from the fact that the initial state on the simulation register might not be an eigenstate of $\mathcal{U}$, but a linear superposition

$$|\psi\rangle = \sum_n c_n |\phi_n\rangle. \quad (4)$$

In this case, the algorithm will collapse the wave function over one of the eigenstates, with a probability $|c_n|^2$, and provide the phase estimate corresponding to the eigenstate ϕ_n (see Appendix A for a demonstration).

In general, taking into account both sources of uncertainty, we can show (see Appendix A) that the probability of reading out a given phase $0.\phi$, for the state in Eq. (4), is given by

$$P(0.\phi) = \sum_n |c_n|^2 P_{\phi_n}(0.\phi). \quad (5)$$

These two sources of uncertainty make it necessary to run the PEA several times and analyze the relative frequency of the $0.\phi$ readouts, $P_\phi(0.\phi)$. In this work, we only consider the case of preparing $|\psi\rangle$ as an eigenstate of the $\mathcal{H}$ operator. This is the building block for understanding the simulation of superposition states.

B. Circular statistics

Consider the execution of the PEA for a given Hamiltonian, $\mathcal{H}$, using a fixed initial eigenstate $|\phi\rangle$, so that the eigenvalue of $\mathcal{U}$ is given by $e^{i2\pi\phi}$. The first thing to note from Eq. (3) is that the final statevector on the phase register subspace perfectly encodes ϕ, and it can in principle be estimated from a sample with adequate confidence intervals, even with $R = 1$. However, this may not be a feasible approach in the presence of noise,

when the distribution is unknown, making it necessary to explore different estimators.

Let us take two different ways of inferring ϕ from this parent distribution. The first one would be to identify ϕ with the $0.\phi$ that has the largest $P_\phi(0.\phi)$, that is $\hat{\phi} = \widetilde{0.\phi}$ for $P_\phi(\widetilde{0.\phi}) \geq 4/\pi^2$. This biased estimator is the *majority* rule most often used in the literature [9, 11, 20, 21], for which the absolute accuracy error is bounded by $\epsilon < 2^{-(R+1)}$, where

$$\epsilon = |\hat{\phi} - \phi|. \tag{6}$$

The alternative way is to take an average. For that matter, we define the *first trigonometric moment about the mean direction* [40] as

$$\theta_1 = \sum_{0.\phi} P_\phi(0.\phi)e^{i2\pi 0.\phi} \equiv \rho e^{i2\pi\mu}, \tag{7}$$

which using Eq. (3) computes to

$$\theta_1 = \frac{2^R - 1}{2^R}e^{i2\pi\phi} + \frac{1}{2^R}e^{-i(2^R-1)2\pi\phi}. \tag{8}$$

Here, μ is the *mean phase direction*, given by

$$\mu(\phi) = \frac{1}{2\pi}\arctan\left(\frac{A\sin(2\pi\phi) - \sin(A2\pi\phi)}{A\cos(2\pi\phi) + \cos(A2\pi\phi)}\right), \tag{9}$$

with $A \equiv 2^R - 1$, properly piecewise defined such that $\mu(\phi) \in [0,1[$ is continuous. The modulus of θ_1,

$$\rho = \sqrt{4^{-R}(4^R - 2^{R+1} + 2 + 2A\cos(2^{R+1}\pi\phi))}, \tag{10}$$

is called the *mean resultant length* and provides a measure of how narrow this unimodal distribution is. It can be translated to a more familiar form to non-circular statistics by defining the *phase circular standard deviation* as

$$\sigma = \frac{\sqrt{-2\ln\rho}}{2\pi}. \tag{11}$$

We can assign a geometric interpretation to Eq. (7): if we think of each complex number in the sum as a vector in the complex plane, ρ stands for the length of the resulting vector, and μ describes its orientation. A uniform distribution would yield a vanishing ρ.

From this approach, we can define a new estimator $\hat{\phi} = \mu$. The maximum accuracy error can be obtained straightforwardly and is seen to be bounded by $\epsilon < 2^{-(R+2)}$ for $R \geq 2$, decreasing monotonically with R. Therefore, this is again a biased estimator for general ϕ but significantly improves on the previous rule. Specifically, the estimation of ϕ from the mean phase direction using R qubits is below the accuracy error bound for the majority rule estimator with $R + 1$ qubits. Therefore, in this sense, the estimation of the phase using the average permits to decrease R by one unit, which makes circuits shallower by avoiding one exponentiation of $\mathcal{U}$.
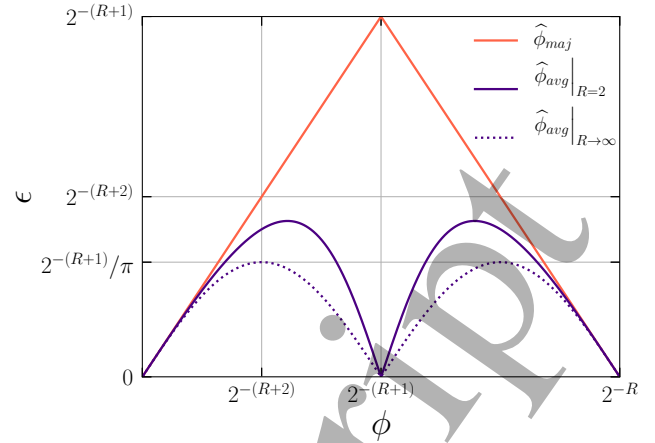

FIG. 2: Modulus of the accuracy error, defined in Eq. (6), as a function of the phase ϕ for two different phase estimators $\hat{\phi}$: the maximum of Eq. (3) (orange) and the mean phase direction of the distribution obtained with Eq. (9) (purple). R stands for the number of register qubits. All the curves for $\hat{\phi}_{avg}$ with $R > 2$ lie between the filled and the dashed purple lines.

In Fig. 2, we compare the absolute value of the bias for the majority and average rules as we ramp ϕ between two adjacent values of $0.\phi$. It is apparent that ϵ is always *smaller* for the average rule. This is in part due to a cancellation of errors: when the phase ϕ is right in the middle, the average rule gives an exact estimation. Interestingly, an asymptotic limit is hit for infinite resolution, $R \to \infty$, in which the error bound scales as $2^{-(R+1)}/\pi$.

We also note that Eq. (9) reveals there is a one-to-one map between ϕ and μ, such that Eq. (3) can be reparametrized by the mean phase direction. Given that μ is simple to estimate from the sample, inverting Eq. (9) provides an unbiased estimator for ϕ at any $R \geq 2$. This allows sparing increasing executions of the unitary and trade them by sampling, which represents a meaningful improvement for the execution of QPE algorithms on devices with short coherence time, such as NISQ hardware.

Using any of these estimators for ϕ would be useful if we were only interested in determining the phase of the propagator, however, we want to get to the eigenvalues of the Hamiltonian. For that, we perform several experiments with different τ to employ Eq. (1), and in this case, we can just as well use μ directly without the additional inversion step.

C. Sampling

Running the PEA in a fault-tolerant quantum computer requires taking a finite sample of O observations from the parent distribution in Eq. (3) to infer its single parameter: either ϕ or μ if the distribution is reparametrized. We can use the sample mean phase direction, $\overline{\mu}$, as an unbiased estimator of μ given by Eq. (9).
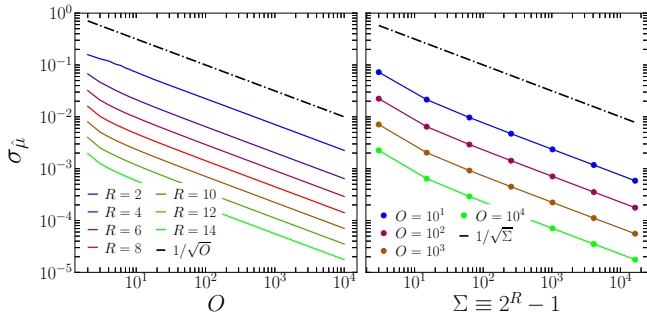
FIG. 3: Numerical predictions for $\sigma_{\hat{\mu}}$ obtained with $S = 10^7$ samples drawn from Eq. (3) at $\phi = 2^{-(R+1)}$, where dispersion is maximal. Most other ϕ values yield $\sigma_{\hat{\mu}}$ lower than these bounds. On the left panel, the dependence of $\sigma_{\hat{\mu}}$ on O is traced for different values of R. The right panel shows how increasing R reduces $\sigma_{\hat{\mu}}$ while fixing the number of observations. Σ represents the total number of $\mathcal{U}$ executions throughout the PEA. Recall that μ is different for each R.

If the same experiment could be repeated with S samples of O observations each to compute $\overline{\mu}$ in each one, we would approximate the sampling distribution of $\hat{\mu}$ for the chosen value of O. This could in turn be used to calculate $\sigma_{\hat{\mu}}$, given by the circular standard deviation of the S sample mean phase directions $\overline{\mu}$.

In the left and right panels of Fig. 3, we verify scaling of $\sigma_{\hat{\mu}}$ with the inverse square root of O and Σ, respectively. For the case of the right panel, $\sigma_{\hat{\mu}}$ can actually decrease faster when the exponentiation of the unitary is implementable with fewer executions. As we show below for the Zeeman and Ising Hamiltonians, the exponentiation of these unitaries can be implemented with a single parametrized execution using the IBM Q gate set.

Quantifying the standard error of $\hat{\mu}$ in this way requires a large quantum computational overhead. Fortunately, there are less computationally intensive ways of doing so, for instance through bootstrapping [41]. This approach can be used straightforwardly with a fault-tolerant quantum computer to quantify the dispersion of $\hat{\mu} = \overline{\mu}$ and, consequently, its *mean squared error*

$$\text{MSE}\,(\hat{\mu}) = \sigma_{\hat{\mu}}^2 + \text{bias}_{\hat{\mu}}^2, \tag{12}$$

since the bias

$$\text{bias}_{\hat{\mu}} = \langle \hat{\mu} \rangle - \mu \tag{13}$$

would be zero in these computers.

However, in state-of-the-art machines, the experimental distribution of results does not approach the unitary one because of the introduction of errors which can generally add both dispersion and skewness to Eq. (3). Since bootstrapping only requires the sequence of observations to be independent and identically distributed, we can still quantify dispersion of the estimator with this technique if the noise process is stationary. This is useful for making no assumptions on a noise model besides this one.

Even so, dispersion is no longer enough to account for the MSE $(\hat{\mu})$ because the noise fluctuations can bias $\hat{\mu}$, frequently by a much larger magnitude than $\sigma_{\hat{\mu}}$.

On that account, for the results obtained in state-of-the-art devices, we consider the $0.\phi$ phase circular standard deviation of the sample as the error-bar for the $\hat{\mu}$ estimation. This choice is an overestimation of the experimental MSE $(\hat{\mu})$, but it is a safe strategy. As we explain below, we can improve accuracy and precision of the final eigenvalue estimation by performing several experiments with different values of the control parameter τ in the propagator and fitting the results to the expected theoretical model.

D. Iterative PEA

Quantum phase estimation based on the R-qubit quantum Fourier transform can lead to a very high number of gates due to the need to implement the R powers of $\mathcal{U}$ in a single long-depth circuit. To avoid this, we can separate the simulation of each of these powers into different circuits and perform QPE with only an ancillary qubit and a R-fold iterative procedure. This measurement-based approach is made possible by the semiclassical quantum Fourier transform [22] and is known as the iterative phase estimation algorithm (IPEA).

In the IPEA, each digit of one $0.\phi$ observation is estimated with a dedicated measurement using the circuits in Fig. 4. For that, we first fix a desired resolution R in Eq. (2) and adopt one of two exploration procedures for choosing between the intermediate measured states produced by these circuits: *exhaustive* or *non-exhaustive*, explained below.

To begin with, we perform a given number of executions of the circuit in Fig. 4(a) to obtain the relative measurement frequencies of the two basis states of the ancillary qubit. These are used to decide on the b_R bit, in accordance with the adopted exploration procedure. Having fixed this digit, we start an iterator variable k decreasing from $R - 1$ to 1. For each k, the circuit in Fig. 4(b) is repeatedly executed with

$$\omega_k = -2\pi \sum_{l=2}^{R-k+1} \frac{b_{k+l-1}}{2^l} \tag{14}$$

and the ancillary qubit is measured to fix b_k, again from the obtained histogram. In this way, we determine all the bits of Eq. (2) from the least to the most significant, at each iteration performing the z-rotation with an angle that is a function of all the previously measured bits.

The two different modes of operating the IPEA differ in the exploration policy of the state space of the binary tree encoding all the $0.\phi$ decompositions of the phase. With *exhaustive* IPEA, we can probe the probability mass function from Eq. (3) over all the 2^R binary strings, just like with the PEA. Therefore, this exploration mode allows
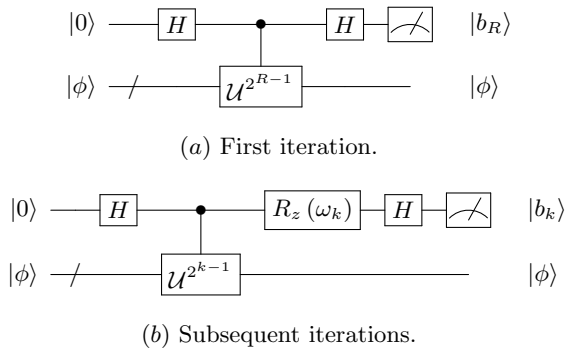
(a) First iteration.

(b) Subsequent iterations.

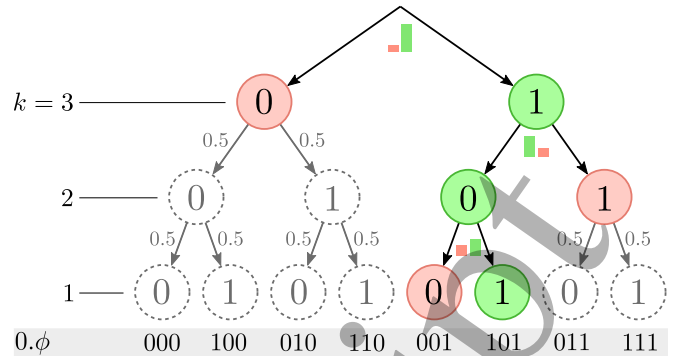FIG. 4: Quantum circuits used by the iterative phase estimation algorithm.

FIG. 5: Reconstruction procedure of the LPMF over all binary strings performed with non-exhaustive IPEA, illustrated for the case of $R = 3$. Only one branch of the full binary tree is explored in this algorithm. Green nodes are the digits obtained from a majority rule at each iteration, which encode the final estimation. Branches below the red nodes are unexplored.

us to also employ Eq. (9) when initializing the algorithm with $\mathcal{H}$ eigenstates, as well as to explore general states.

Sometimes, we are only interested in obtaining the value $\widetilde{0.\phi}$ for which the probability of Eq. (3) is maximal. For that, we can perform a *non-exhaustive* exploration of the state space using the IPEA as implemented by Dobšíček et al. [23]. This procedure consists in running each iteration a sufficient number of times to let probabilities converge, choose the most frequently obtained bit as b_k and proceed to the next iteration. The final estimation $\hat{\phi} = \widetilde{0.\phi}$ is the binary string immediately constructed with the most frequent bits obtained in all the iterations. This procedure is equivalent to the PEA with the majority rule. Since this only explores one branch of the full binary tree of the R-bit string (see Fig. 5), it cannot be used to obtain complete information of input superpositions of $\mathcal{U}$ eigenstates.

We again emphasize that our discussion of the mean phase direction estimator is directly applicable to the exhaustive IPEA. However, in the experiments here reported, we only use the non-exhaustive version of the IPEA for the reasons explained in section VII. In this scenario, we can take into account all the 2-state histograms at the intermediate steps to construct the sample approximation to a lossy version of the full probability mass function given by Eq. (3) over all the possible R-bit strings. To approximate this lossy probability mass function (LPMF), we organize the relative frequencies of measurement obtained for each bit in a binary tree format where a single full-depth branch encodes $\widetilde{0.\phi}$ and the unexplored bit measurements correspond to chopped nodes at every level.

Then, the relative frequencies of the unexplored nodes are uniformly propagated down to their descendant leaves in the tree to reconstruct the sample approximation to the LPMF, which in turn approximates Eq. (3). This LPMF distribution cannot provide sufficient information to discern the complexity of Eq. (3) across all the different R-bit states, but allows computing the phase circular standard deviation with Eq. (11), used as a measure of dispersion of the sample obtained with non-exhaustive

IPEA. Using Eq. (9) over the LPMF cannot improve accuracy.

The IPEA has the big advantage of achieving a very large reduction of the gate count while also overcoming the limit in resolution imposed by the number of qubits used to register the phase in the PEA. The only significant increase in computing time comes from preparing the initial state at every run.

E.   Least squares regression

In order to determine a given eigenvalue, we use Eq. (1). To do so, we carry out simulations with different values for τ and extract the slope by fitting the phase estimates $\hat{\phi}$ to a theoretical model $f(\tau)$. In the case of the majority rule estimator, we take $f(\tau) = m\tau + b$ because even if this is not the exact model, there is a linear dependence of $\hat{\phi}$ in τ. In the case of the average rule estimator, we use $f(\tau) = \mu(m\tau + b)$, where μ is given by Eq. (9). However, given the periodic nature of the data, which causes problems when $\hat{\phi}$ is close to 0 and 1, our approach is to use the least squares fitting method to minimize a modified χ^2 functional given by

$$\chi^2_{\text{circ}} = \sum_{i=1}^{N} \frac{|e^{i2\pi\hat{\phi}_i} - e^{i2\pi f(\tau_i)}|^2}{\sigma_i^2}, \qquad (15)$$

where σ_i is the error-bar associated to the estimate for τ_i. The fitting procedure determines the parameters m and b. This functional can be split into two,

$$\chi^2_{\text{circ}} = \chi^2_{\cos} + \chi^2_{\sin}, \qquad (16)$$

where

$$\chi^2_{\cos} = \sum_{i=1}^{N} \left(\frac{\cos(2\pi\hat{\phi}_i) - \cos(2\pi f(\tau_i))}{\sigma_i} \right)^2, \qquad (17)$$

$$\chi^2_{\sin} = \sum_{i=1}^{N} \left(\frac{\sin(2\pi\hat{\phi}_i) - \sin(2\pi f(\tau_i))}{\sigma_i} \right)^2. \quad (18)$$

Thus, minimizing the functional in Eq. (15) is equivalent to using the standard least squares method to minimize $\chi^2_{\cos}$ and $\chi^2_{\sin}$ simultaneously.

IV.　SPIN HAMILTONIANS

In order to test our DQS and QPE methods, we first apply them to spin-1/2 Hamiltonians. For such problems, the eigenstates of the Hamiltonian are identical to the computational basis. Here, we introduce the two spin models considered in this work and show explicitly how to implement the unitary evolution operators using one- and two-qubit gates.

A.　Two-level system

The simplest Hamiltonian one could think of is a two-level system with energy splitting ω,

$$\mathcal{H}_Z = \omega Z, \quad (19)$$

where Z is the Pauli Z operator. The Hilbert space has dimension 2 and can therefore be encoded in a single qubit. In the spin language, this model can be interpreted as the Zeeman Hamiltonian of a $S = 1/2$ spin. The energy levels of $\mathcal{H}_Z$ are $\pm\omega$.

For the two-level system, we take the qubit computational basis as the eigenstate basis. Therefore, the *initialization* is straightforward. The unitary evolution operator can be simulated in the quantum computer using the phase shift gate as

$$\mathcal{U}_Z(\tau) = e^{-i\mathcal{H}_Z\tau} = R_z(\theta) = \begin{pmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{pmatrix}, \quad (20)$$

where $\theta = 2\omega\tau$. However, to control the action of $\mathcal{U}_Z(\tau)$, we need to implement two-qubit gates. In Fig. 6, we show how to decompose a controlled-$R_z(\theta)$ operation in terms of primitive gates.
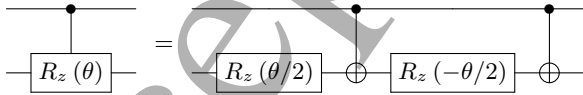

FIG. 6: Gate decomposition of the controlled-$R_z(\theta)$ operator. This circuit directly implements the controlled propagator for the two-level system.

B.　Ising dimer

Our second model moves one step up in complexity scale and accounts for 2 spins with Ising-like interactions that preserve the spin operator S_z. The Hamiltonian reads as

$$\mathcal{H}_I = \omega_1 Z_1 + \omega_2 Z_2 + \omega_J Z_1 Z_2, \quad (21)$$

where the first two terms correspond to decoupled Zeeman Hamiltonians for the two spins, whereas the third term accounts for a Ising-like exchange interaction with energy ω_J.

The four eigenvalues of this Hamiltonian are $\varepsilon_{s_1,s_2} = s_1\omega_1 + s_2\omega_2 + s_1 s_2\omega_J$, where $s_{1,2} = \pm 1$. As in the case of the two-level system, the eigenstates of $\mathcal{H}_I$ can also be encoded in the computational basis of the quantum computer. This is done with a trivial mapping between the state with eigenvalue ε_{s_1,s_2} and the qubit state $|\frac{1-s_1}{2}, \frac{1-s_2}{2}\rangle$.

The Hamiltonian for the Ising dimer, Eq. (21), is a sum of mutually commuting terms, so that its unitary evolution operator can be decomposed as a product of operators corresponding to each term. In the quantum computation language, it can be written as

$$\mathcal{U}_I(\tau) = e^{-i\mathcal{H}_I\tau} = R_z^{(1)}(\theta_1) R_z^{(2)}(\theta_2) e^{-i\theta_3 Z_1 Z_2/2}, \quad (22)$$

where $\theta_1 = 2\omega_1\tau$, $\theta_2 = 2\omega_2\tau$ and $\theta_3 = 2\omega_J\tau$. The quantum circuit that implements the controlled-$\mathcal{U}_I(\tau)$ operation is shown in Fig. 7, where the implementation of the term $e^{-i\theta_3 Z_1 Z_2/2}$ is boxed.
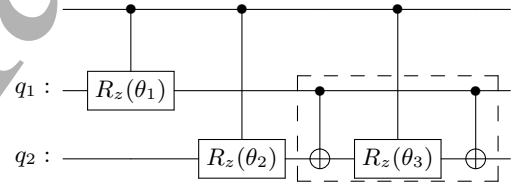

FIG. 7: Quantum circuit to implement the controlled unitary evolution operator for the Ising model. The gate decomposition of the controlled-$R_z(\theta)$ gates is shown in Fig. 6.

V.　RESULTS FOR THE SPIN HAMILTONIANS

We now present the experimental results for the spin models described in the previous section. All digital quantum simulations were carried out on the IBM Q hardware. In this platform, single-qubit operations are implemented with just three physically parametrized gates,

$$U_1(\lambda) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\lambda} \end{pmatrix},$$

$$U_2(\phi, \lambda) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -e^{i\lambda} \\ e^{i\phi} & e^{i(\lambda+\phi)} \end{pmatrix}, \quad (23)$$

$$U_3(\theta, \phi, \lambda) = \begin{pmatrix} \cos(\theta/2) & -e^{i\lambda}\sin(\theta/2) \\ e^{i\phi}\sin(\theta/2) & e^{i(\lambda+\phi)}\cos(\theta/2) \end{pmatrix},$$
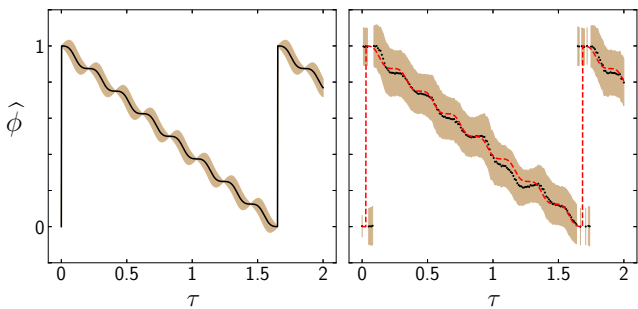
FIG. 8: Results for the PEA implementation of the two-level system. On the left-side panel, the black line traces the mean phase direction parameter of the parent distribution in Eq. (3) as given by Eq. (9) with $\phi = -\varepsilon\tau/(2\pi)$; the brown region enclosing μ corresponds to 1σ, computed with Eq. (11). On the right-side panel, the experimental results obtained in *ibmq_20_tokyo* are shown. The black dots plot $\overline{\mu}$ while the brown region enclosing each point is one experimental σ. The dashed red line is given by $f(\tau) = \mu(m\tau + b \pmod 1)$, where $m = -0.6041 \pm 0.0039$ and $b = 0.0174 \pm 0.0046$ are fitting parameters. The corresponding χ^2 functional per number of degrees of freedom is $\chi^2_{\mathrm{circ}}/\mathrm{ndf} = 0.16$.

which together with the cX gate, form a universal gate set only limited by noise and decoherence. In particular, our implementations use the *ibmq_20_tokyo* 20-qubit device. Even though this machine offers 20 qubits, they are not fully connected, meaning that we cannot make direct cX operations between any two arbitrary qubits. This is an important issue in NISQ computing, as it can greatly increase the gate count. For that reason, for both the two-level system and the Ising dimer, we took a subset of 4 fully-connected qubits that *ibmq_20_tokyo* offers. This allowed us to end up with quantum circuits with small enough depth, for which the PEA can still be employed. Naturally, this choice imposes a limited resolution. For the two-level system, we only need 1 qubit to simulate time evolution and we are left with $R = 3$ qubits in the phase register. For the Ising dimer, the simulation register requires 2 qubits and therefore we can only use $R = 2$ qubits to register the phase. Given this relatively small resolution in the PEA, we compare the experimental results with those from the ideal distribution expected as in Eq. (3).

In Fig. 8, we show the results for the mean phase direction and the phase circular standard deviation obtained for the two-level system, fixing $\omega = 3.8$ and the initial state as $|\psi\rangle = |0\rangle$, whose eigenvalue is $\varepsilon = \omega$. The total number of gates in this circuit is 29, of which 12 are cX. We perform 200 experiments for different values of $\tau \in [0,2]$, in each one taking $O = 8192$ shots. We observe that, even with just $R = 3$, the quantum dispersion of the PEA is much smaller than the experimental noise.

For the reasons explained in section III C, we fit the experimental $\overline{\mu}(\tau)$ with σ given by the sample phase circular standard deviation in Eq. (17) and Eq. (18). We

$\lvert u\rangle$	$\lvert 00\rangle$	$\lvert 01\rangle$	$\lvert 10\rangle$	$\lvert 11\rangle$
ε	4.74	−4.08	1.74	−2.40
m	−0.763 ±0.009	0.662 ±0.007	−0.292 ±0.004	0.364 ±0.003
b	−0.078 ±0.007	−0.037 ±0.007	0.018 ±0.007	−0.001 ±0.005
$\chi^2_{\mathrm{circ}}/\mathrm{ndf}$	0.37	0.09	0.38	0.28
$\hat{\varepsilon}$	4.79 ±0.06	−4.16 ±0.05	1.83 ±0.03	−2.29 ±0.02
$(\varepsilon - \hat{\varepsilon})/\lvert\varepsilon\rvert$	−0.01	0.02	−0.05	−0.05
$(\varepsilon - \hat{\varepsilon})/\delta\hat{\varepsilon}$	−0.8	1.6	−3.0	−5.5

TABLE I: Fitting parameters for the experimental results of the PEA with the Ising dimer.

find a good agreement between experimental and theoretical results, obtaining an estimation for the eigenvalue of $\hat{\varepsilon} = 3.80 \pm 0.02$. The fact that $b = 0.0174 \pm 0.0046$ does not cover the expected value $b = 0 \bmod 1$ is interpreted as a phase shift error that can be due to calibration, for instance.

In the case of the Ising dimer, we explore the initialization of the system in its four eigenstates, $|00\rangle$, $|01\rangle$, $|10\rangle$ and $|11\rangle$. These are explored in separate, though, and we do not address here the linear superposition case. For computation purposes, we take $\omega_1 = 0.33$, $\omega_2 = 3.24$ and $\omega_J = 1.17$. The results, obtained with $O = 5000$ shots, are shown in Fig. 9 and Table I. Despite the different initialization of each eigenstate, which in this case takes at most two X gates for the preparation of $|11\rangle$, all the PEA circuits for the Ising dimer are compiled by the platform to yield 35 gates, of which 18 are cX. Once again, we observe that the dispersion due to finite R is much smaller than the experimental noise. For this model Hamiltonian, the increased complexity leads to experimental values that do not cover the theoretical ones. Even so, the relative errors are small and the standard score is roughly between 1-3 (except for the $|11\rangle$ eigenstate), meaning that the experimental values fall off from the theoretical ones by 1-3 error bars. Therefore, we verify that the PEA can handle the Ising dimer. Finally, we note that for three out of four initial states, we get values of b that indicate the presence of phase shift errors. In particular, when the state is initialized as $|00\rangle$, this phase shift is evident just by comparing the top plots of Fig. 9. The phase shift error, though, does not affect the estimation of the energies.

VI. HUBBARD MODEL

We now consider a 2-site Hubbard model, whose Hamiltonian reads as

$$\mathcal{H}_{\mathrm{H}} = U \sum_{i=a,b} n_{i\uparrow} n_{i\downarrow} - t \sum_{\sigma=\uparrow,\downarrow} \left(a^\dagger_\sigma b_\sigma + b^\dagger_\sigma a_\sigma \right), \qquad (24)$$
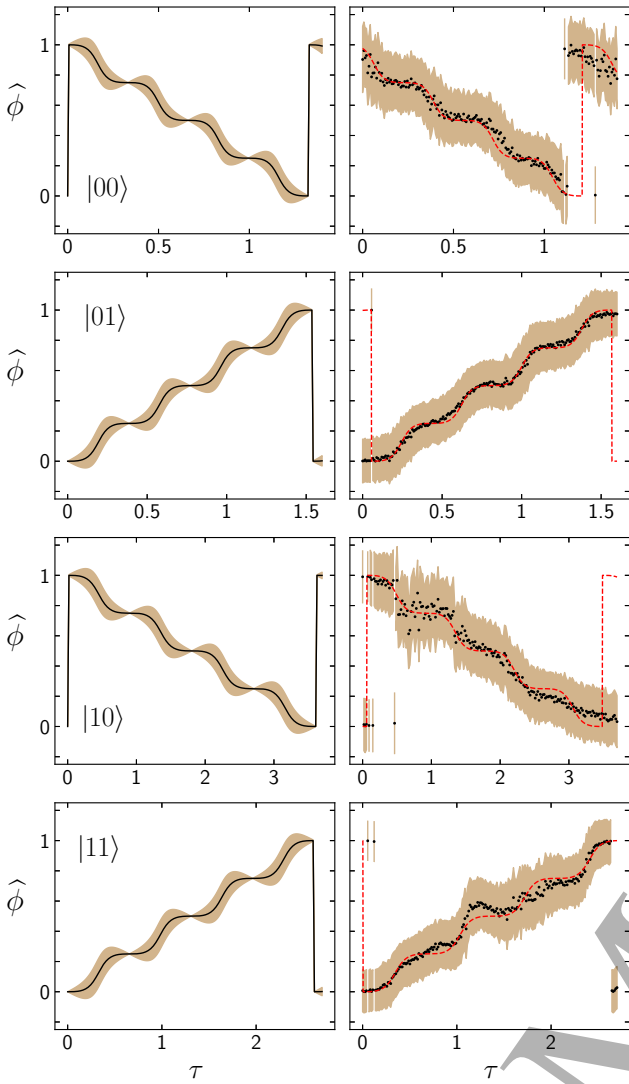
FIG. 9: PEA results for the simulation of the four eigenstates of the Ising dimer. The left-side column plots μ (black) and 1σ (brown) derived from the parent distributions of fault-tolerant circuits, as in the case of the Zeeman simulation. The right-side panels show *ibmq_20_tokyo* results: $\overline{\mu}$ is represented by the black dots and the experimental disperson of 1σ is marked in brown; dashed red lines show the fit results of $f(\tau) = \mu(m\tau + b \pmod{1})$ to this data, with m and b as parameters (see Table I).

where U, t are positive parameters, a_σ/b_σ is the fermionic annihilation operator for an electron in site a/b with spin $\sigma = \uparrow, \downarrow$ and $n_{a\sigma} = a_\sigma^\dagger a_\sigma, n_{b\sigma} = b_\sigma^\dagger b_\sigma$ are the corresponding number operators.

The DQS of the Hubbard model poses several additional challenges missing in the case of the two spin models. First, we need to map fermions to qubits. Second, the model is a sum of non-commuting terms, which complicates the simulation of the unitary evolution. Third, the preparation of the initial states is no longer trivial.

A. From fermions to qubits: Jordan-Wigner transformation

We use a Jordan-Wigner transformation [42] to map fermionic operators into qubit operators. This requires one qubit per spin-site, which leads to 4 qubits in the case of the Hubbard dimer. We associate the states $a \uparrow$, $b \uparrow$, $a \downarrow$, $b \downarrow$ (whose occupation can be either 0 or 1) to the qubits $1, 2, 3, 4$, respectively. By doing so, the Hubbard Hamiltonian reads as $\mathcal{H}_\mathrm{H} = \mathcal{H}_U + \mathcal{H}_t$, with

$$H_U = \frac{U}{4}\left(2\mathbf{1} + Z_3 Z_1 + Z_4 Z_2 - \sum_{i=1}^{4} Z_i\right), \quad (25)$$

$$H_t = -\frac{t}{2}\left(X_2 X_1 + Y_2 Y_1 + X_4 X_3 + Y_4 Y_3\right), \quad (26)$$

where X (Y) denotes the Pauli X (Y) matrix, that acts on the qubit subspace labeled by the subscript.

The Hubbard dimer is thus mapped into a spin model with 11 terms that, unlike the case of the Ising dimer in Eq. (21), are non-commuting. This brings an additional layer of complexity when it comes to simulate the unitary evolution. The quantum circuit to implement the PEA with this Hamiltonian, for just one 1st-order Trotter step, with 3 register qubits and in a quantum computer with 3+4 fully-connected qubits, already has close to 500 gates in total, of which over 200 are cX gates. Moreover, these numbers do not even account for the circuit that needs to be built in order to prepare the initial state. Both of these requirements make its practical implementation in current hardware impossible. Therefore, we adopt a different strategy, which requires a much smaller number of gates.

B. Hubbard dimer at half-filling: compact representation

At half-filling, the Hilbert space of the Hubbard dimer has 6 states. Two of them have spin projection $s_z = +1, -1$ and, since the spin operator S_z commutes with the Hamiltonian, these are decoupled from the remaining four states, that have $s_z = 0$. Thus, it is possible to find a compact representation [43] for the non-trivial states of the half-filled Hubbard dimer that makes use of only 2 qubits. The four non-trivial states, written in the basis $\{a \uparrow, b \uparrow, a \downarrow, b \downarrow\}$, are mapped into qubit states as

$$\{1, 0, 1, 0\} \equiv \uparrow\downarrow \bigcirc = |00\rangle$$
$$\{0, 1, 1, 0\} \equiv \downarrow \ \uparrow = |01\rangle$$
$$\{1, 0, 0, 1\} \equiv \uparrow \ \downarrow = |10\rangle$$
$$\{0, 1, 0, 1\} \equiv \bigcirc \uparrow\downarrow = |11\rangle. \quad (27)$$

In this basis, the (reduced) Hamiltonian is given by

$$\mathcal{H}_\mathrm{H}' = \begin{bmatrix} U & -t & -t & 0 \\ -t & 0 & 0 & -t \\ -t & 0 & 0 & -t \\ 0 & -t & -t & U \end{bmatrix}. \quad (28)$$

Now, we can encode this Hamiltonian using 2 qubits, 1 and 2, that act on the spin $\downarrow$ and $\uparrow$ electrons, respectively. Inspection of Eq. (27) shows that that X_1, for instance, carries out the hopping the $\uparrow$ electron, whereas the operator Z_1 measures in which of the two Hubbard sites it is located. Using these operators, we can write up the Hamiltonian in Eq. (28) as

$$\mathcal{H}'_{\mathrm{H}} = -t(X_1 + X_2) + \frac{U}{2}(Z_1 Z_2 + \mathbf{1}) \qquad (29)$$

where $X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ is the Pauli X operator. This permits to model the two-site Hubbard model at half-filling using only 2 qubits. After this reduction procedure, it must be noted that the Pauli operators of the Hamiltonian in Eq. (29) are not directly related to the original fermions via a Jordan-Wigner transformation. It must also be noted that Eq. (29) can represent a 2-site Ising model with transverse magnetic field.

The energy spectrum of Eq. (28) is given by

$$\varepsilon_\pm = \frac{U}{2} \pm \sqrt{4t^2 + U^2/4},$$
$$\varepsilon_b = 0 \quad, \varepsilon_c = U \qquad (30)$$

In the $t = 0$ limit, these states have energies $\varepsilon_+ = U = \varepsilon_c$, $\varepsilon_- = \varepsilon_b = 0$. In the $U = 0$ limit, we have $\varepsilon_b = \varepsilon_c = 0$ and $\varepsilon_\pm = \pm 2t$. The corresponding eigenstates are

$$|\varepsilon_b\rangle = \frac{1}{\sqrt{2}} (|10\rangle - |01\rangle), \qquad (31)$$

$$|\varepsilon_c\rangle = \frac{1}{\sqrt{2}} (|11\rangle - |00\rangle), \qquad (32)$$

$$|\varepsilon_\pm\rangle = \alpha_\pm [|00\rangle + |11\rangle + \beta_\pm (|10\rangle + |01\rangle)], \qquad (33)$$

where

$$\alpha_\pm = \left(\frac{\left(U \mp \sqrt{16t^2 + U^2}\right)^2}{8t^2} + 2 \right)^{-1/2}, \qquad (34)$$

$$\beta_\pm = \frac{U \mp \sqrt{16t^2 + U^2}}{4t}. \qquad (35)$$

Therefore, they are no longer simple product states in the computational basis. As a result, the preparation of the eigenstates is no longer straightforward and we need to craft a procedure to initialize these states. We use the algorithm by Shende et al. [39] to do so (see Fig. 12).

C. Trotter-Suzuki formulas

In general, the Hamiltonian of most interacting systems consists in a sum of non-commuting terms. This is the case of the Hubbard Hamiltonian in Eq. (29), which is the sum of two terms that do *not* commute with each other. Since we would like to simulate the time propagator by exponentiating each Hamiltonian term individually (Appendix B) and multiplying them together, we can use the Trotter-Suzuki approximations. These allow us to express the exponential of a sum of non-commuting operators in terms of products of unitaries. To do so, each group of commuting terms is exponentiated individually for a small time-step and the exponentials are multiplied in such a way as to provide a reasonable approximation of the time-evolution operator. The first and second-order Trotter-Suzuki expansions are given by [44, 45]

$$e^{(A+B)\tau} = \left(e^{A\tau/n} e^{B\tau/n} \right)^n + \xi_1, \qquad (36)$$

$$e^{(A+B)\tau} = \left(e^{A\tau/2n} e^{B\tau/n} e^{A\tau/2n} \right)^n + \xi_2, \qquad (37)$$

where $[A, B] \neq 0$, and we have errors $\xi_1 = O(\tau \Delta \tau)$ and $\xi_2 = O(\tau (\Delta \tau)^2)$. Since A commutes with itself, Eq. (37) can be recast into the product of fewer exponentials as

$$e^{(A+B)\tau} = e^{\frac{A\tau}{2n}} \left(e^{\frac{B\tau}{n}} e^{\frac{A\tau}{n}} \right)^{n-1} e^{\frac{B\tau}{n}} e^{\frac{A\tau}{2n}} + \xi_2. \qquad (38)$$

The approximation is improved by increasing n, the *Trotter number*, thus making the time step $\Delta \tau = \tau/n$ small compared to the smallest time scale of $\mathcal{U}(\tau)$. As more repetitions are made, or equivalently as $\Delta \tau \to 0$, the error vanishes. However, there is a trade-off: making $\Delta \tau$ small, requires increasing the number of steps, and thereby the number of gates, which leads to errors when the algorithm is implemented in noisy hardware.

A good approximation is also possible if we take the evolution time to be much smaller than the smallest time scale of the operator. In the limit $\tau \to 0$, the errors in Eq. (36) and Eq. (38) scale asymptotically as

$$\xi_1 \stackrel{\tau \to 0}{\sim} \frac{\tau \Delta \tau}{2} [B, A], \qquad (39)$$

$$\xi_2 \stackrel{\tau \to 0}{\sim} \frac{\tau (\Delta \tau)^2}{24} ([A, [A, B]] + 2 [B, [A, B]]). \qquad (40)$$

In QPE, we need to simulate $\mathcal{U}(\tau)$ raised to increasing powers of 2. This is done by absorbing the exponent into τ and implementing $\mathcal{U}^{2^{k-1}}(\tau) = \mathcal{U}(\tau 2^{k-1})$, in which case we have $\xi_1 = O(\tau \Delta \tau 4^{k-1})$ and $\xi_2 = O(\tau (\Delta \tau)^2 8^{k-1})$, and the right-hand sides of Eq. (39) and Eq. (40) are multiplied by 4^{k-1} and 8^{k-1}, respectively. Due to this dependence of the error in the QPE exponent, we should let $n \to n2^{k-1}$ each time the propagator is exponentiated, both if we use the first- or the second-order decompositions. Increasing n in this way, we cancel higher orders of the exponent in the error and guarantee it increases only linearly with the simulated exponent, achieving the required $\xi_1 = O(\tau \Delta \tau 2^{k-1})$ and $\xi_2 = O(\tau (\Delta \tau)^2 2^{k-1})$. Thus, circuit depth increases exponentially with R.

We are now in a position to discuss the quantum circuit that implements the time evolution operator of the Hubbard dimer. First, we shift the Hamiltonian in Eq. (29) to drop the constant term, which leaves us with only three terms to exponentiate. The new eigenvalues are given in terms of the original ones in Eq. (30) by $\varepsilon_i' = \varepsilon_i - U/2$. Second, we separate the non-commuting terms of the Hamiltonian into two sets and implement each of these controlled propagators as in Fig. 10. Applying these circuits with the right ordering and rotation parameters builds the desired Trotter order and number. We choose to perform our experiments by implementing the Trotter approximated propagator $\tilde{\mathcal{U}}(\tau)$ with the first order formula in Eq. (36) and $n = 1$, in the $\tau \to 0$ limit.
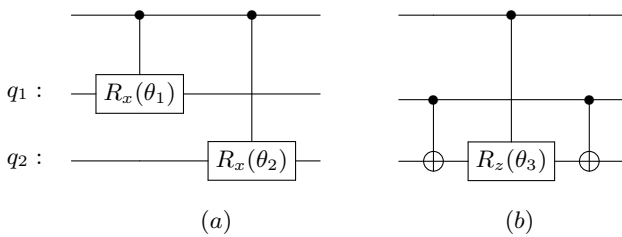

FIG. 10: Controlled implementation of the propagators coming from the two groups of commuting terms in Eq. (29). Using the first order approximation, we have $\theta_1 = \theta_2 = -2t\tau/n$ and $\theta_3 = U\tau/n$, while with Eq. (38), θ_3 remains the same but $\theta_1 = \theta_2$ are divided by 2 for the first and last exponentials. As explained, the exponents of the phase estimation algorithms are introduced as 2^{k-1} factors in θ_1, θ_2 and θ_3.
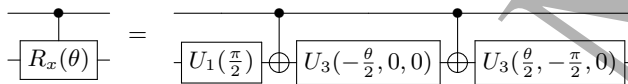

FIG. 11: Implementation of the controlled-$R_x(\theta)$ operation using IBM's platform gate set in Eq. (23).

VII. RESULTS FOR THE HUBBARD DIMER

Let us now discuss the QPE results for the DQS of the Hubbard dimer at half-filling. For the purposes of testing our methodology, we focus on the groundstate of the model and choose the Hamiltonian parameters in Eq. (29) to be $t = 0.35$ and $U = 0.2$. The procedure is identical for different eigenstates and parameter choices. The theoretically expected eigenvalue is thus $\varepsilon_- \approx -0.6071$. To prepare this state, we use the circuit in Fig. 12 to bring simulation register from $|00\rangle$ to $|\varepsilon_-\rangle$.

Due to the stringent circuit depth requirements, we use the IPEA to reduce the noise level in the final computation as much as possible. We choose to carry out the experiments with the non-exhaustive version of the
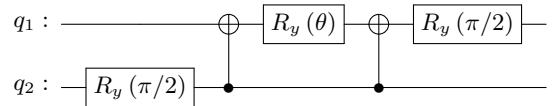

FIG. 12: Initialization circuit for $|\varepsilon_-\rangle$. The exact amplitudes of the groundstate obtained with $t = 0.35$ and $U = 0.2$ are reproduced with $\theta = 0.14189705(...)$, input with the maximum possible numerical precision by the IBM Q platform.

algorithm for two reasons. The first one is logistical: currently, IBM Q does not allow for measurement feedback and state re-setting of the quantum circuits. This makes performing the exhaustive IPEA a lengthy task when R is sufficiently large because of the need to request a high number of circuit executions to the system and going through a communication bottleneck. The second reason is methodological: when Trotterization is needed, we cannot avoid working with a superposition state if we have a fixed initialization procedure across different values of τ. This is because the Trotter approximated propagator, $\tilde{\mathcal{U}}(\tau)$, does not commute with itself at different times and there is no common basis for the operator for all τ. Therefore, if we just want to probe the phase coming from one of the eigenstates (the groundstate in our case) we cannot use the mean phase direction estimator to do so directly and without further post-processing, because the $0.\phi$ frequency counts will include the contribution of nearby eigenstates. The use of the mean phase direction estimator with superposition states is not explored here.

We must now remember that, apart from quantum localization effects that can bound the Trotter error periodically in τ even at long evolution times [46], the approximation is only valid for $\tau \approx 0$ if a small Trotter n is to be used. Hence only short evolution times should be simulated for the obtained phase estimates to be adequate for regression with a linear function $f(\tau) = m\tau + b$ in Eq. (15). Since we want to maximize the number of sampled τ points to improve the quality of the regression, we also simulate negative time to obtain a symmetric time interval close to, and centered at, $\tau = 0$. We perform experiments for 200 points in $|\tau| < 5$.

In Fig. 13 we show the experimental results for 4 levels of resolution, as well as the same computational results obtained in a simulated fault-tolerant quantum computer, for comparison. For each timestamp, $\widehat{\phi}(\tau)$ is plotted with the standard deviation of the reconstructed LPMF represented by the brown region. Every point is the result of R iterations, each one with $O = 5000$ observations. As in the case of the computations in section V, experimental dispersion on the right-hand side column is much larger than what would be expected for ideal quantum computations, deteriorating with increasing R due to the higher vulnerability to noise and decoherence that comes with the increase in circuit depth. Nonetheless, as we describe below, we find a good agreement between theory and experiment when we fit the experimental datapoints to the non-noisy theoretical model. We could
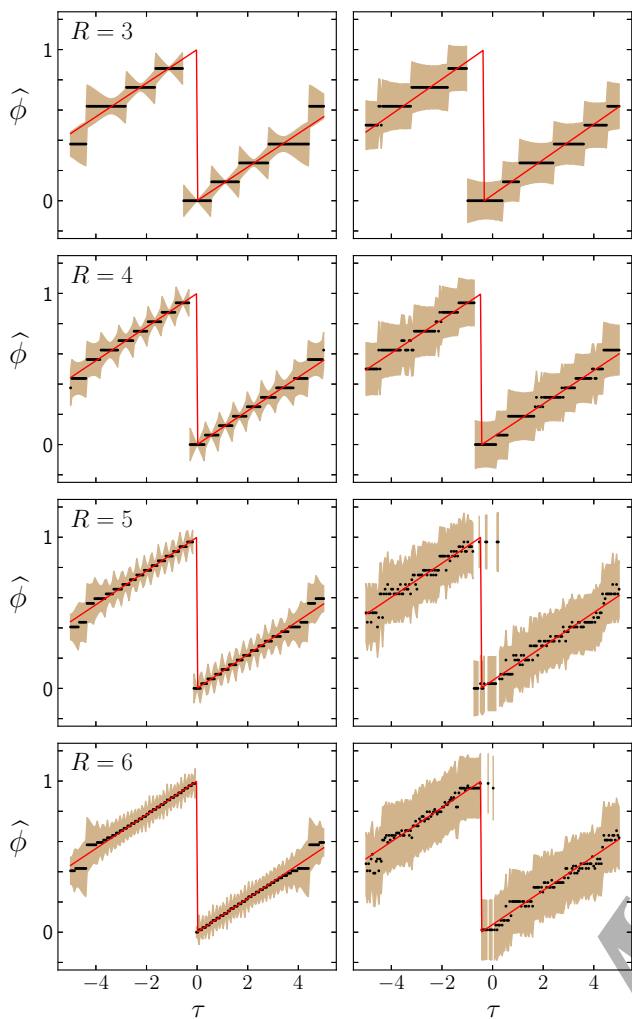
FIG. 13: Unitary simulator (left) and experimental (right) results for the non-exhaustive IPEA with the Hubbard model, using a 1st order Trotter approximation with $n = 1$, and varying resolution from 3 to 6. The fitted line is obtained with the points for $|\tau| < 3$ (see Tables III and II).

not increase R further because with $R = 6$ we already achieved the maximum number of gates that IBM allows in real hardware (see Fig. 14).

To understand these results, one should naturally expect that the initial prepared state on the simulation register, while being an eigenstate of the Hubbard Hamiltonian, is not an eigenstate of $\tilde{\mathcal{U}}(\tau)$, on account of the Trotter approximation. This in turn means that the initial state has a decomposition in the eigenstates of $\tilde{\mathcal{U}}(\tau)$. For short evolution times, the overlap of the initial state with the groundstate of $\tilde{\mathcal{U}}(\tau)$ is large, however, as τ increases, the initial state progressively departs from this eigenket and eventually starts to overlap the most with some other nearby state, causing the maximum peak in the probability function to suddenly jump across basis states, as can be seen around $\tau = 4$.

We choose a smaller interval to fit only the datapoints $|\tau| < \tau_{\max}$, with $\tau_{\max} = 3$. The fitting parameters are displayed in Table III, Table II and Fig. 14, and the fitted model is plotted in red in Fig. 13. We can see that the most accurate results obtained with *ibmq_20_tokyo* are quite good, with a relative error of $\approx 0.2\%$ for $R = 5$ and $\approx 0.4\%$ for $R = 6$. Only these two experiments are more accurate in *ibmq_20_tokyo* than in unitary simulations. This is interpreted as a fortuity statistical occurrence which is actually not significant since the precision error $\delta\hat{\varepsilon}$ of the estimation is consistently smaller in the unitary experiments.

Importantly, unitary simulations show we hit an asymptotic limit where the accuracy error is not tending towards 0, because in the chosen $|\tau| < \tau_{\max}$ interval, the datapoints do not trace a perfectly straight line due to the inaccuracy of the first order Trotter approximation.

This brings us to the most important conclusion learnt from this methodology. Increasing accuracy while reducing the standard score by linearly fitting $\hat{\phi}(\tau)$ requires using the $\tilde{\mathcal{U}}(\tau)$ decomposition that best matches the actual time propagator of the system in the range of simulated τ values. This can be done by either increasing the Trotter order while keeping a fixed $\tau_{\max}$, or by fixing a Trotter order and a Trotter number (in our case, first order and $n = 1$) and limiting regression to shorter evolution times, that is zooming into $\tau = 0$ by reducing $\tau_{\max}$. The challenge posed by the second option is that we start to hit the $0.\phi$ resolution limit for the phase readouts to reliably fit the data, calling for the necessity to increase R simultaneously with the decrease in $\tau_{\max}$ to avoid obtaining a stepwise signal.

Thus, if increasing R increases the number of gates exponentially, it is also true that it increases the number of readout basis states $0.\phi$ exponentially. This allows us to perform an exponential zoom into $\tau = 0$ and get an exponential increase in accuracy, since we beat the scaling of the Trotter error, which is polynomial. The ability to achieve chemical accuracy thus depends on the ability to implement exponentially deeper circuits. The procedure would be to fix initial values for $\tau_{\max}$ and R to perform the first experiment and then, each time R is increased by one unit, $\tau_{\max}$ would be divided by 2.

VIII. DISCUSSION AND CONCLUSIONS

In this work, we have looked at the use of QPE algorithms for the determination of Hamiltonian eigenvalues both from a theoretical and an experimental points of view. We explored the scenario of eigenstate initialization of these algorithms, setting the basis for more general treatments using our methods.

On the first front, we have derived an expression for the first trigonometric moment about the mean direction of the fault-tolerant parent probability distribution associated with eigenstate inputs to full-blown QPE (the PEA and the exhaustive IPEA). This quantity can be inexpen-
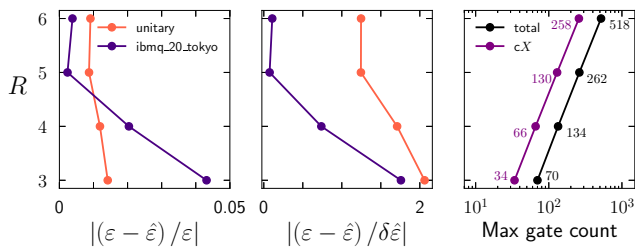
FIG. 14: Relative errors, absolute standard score and number of gates in the most demanding iteration for each non-exhaustive IPEA experiment with the Hubbard dimer. Both the experimental results and those obtained in a unitary classical simulator are shown as a function of the resolution R.

R	3	4	5	6
m	0.1111 ±0.0007	0.1114 ±0.0007	0.1117 ±0.0007	0.1117 ±0.0007
b	0.0000 ±0.0004	0.0000 ±0.0008	0.0002 ±0.0004	0.0001 ±0.0005
$\chi^2_{\rm circ}/{\rm ndf}$	1.63	0.38	0.10	0.03
$\hat{\varepsilon}$	−0.599 ±0.004	−0.600 ±0.005	−0.602 ±0.004	−0.602 ±0.004
$(\varepsilon - \hat{\varepsilon})\,/\,|\varepsilon|$	−0.013	−0.012	−0.008	−0.008
$(\varepsilon - \hat{\varepsilon})\,/\delta\hat{\varepsilon}$	−2.03	−1.42	−1.28	−1.28

TABLE II: Fitting parameters for the unitary results of the non-exhaustive IPEA with the Hubbard dimer.

R	3	4	5	6
m	0.1167 ±0.0024	0.1106 ±0.0027	0.1123 ±0.0031	0.1129 ±0.0034
b	0.0382 ±0.0040	0.0473 ±0.0046	0.0522 ±0.0053	0.0504 ±0.0059
$\chi^2_{\rm circ}/{\rm ndf}$	0.42	0.17	0.18	0.12
$\hat{\varepsilon}$	−0.633 ±0.015	−0.595 ±0.017	−0.606 ±0.019	−0.609 ±0.022
$(\varepsilon - \hat{\varepsilon})\,/\,|\varepsilon|$	0.043	−0.020	−0.002	0.004
$(\varepsilon - \hat{\varepsilon})\,/\delta\hat{\varepsilon}$	1.76	−0.74	−0.07	0.11

TABLE III: Fitting parameters for the experimental results of the non-exhaustive IPEA with the Hubbard dimer.

sively estimated from a sample, without bias, introducing a new avenue for classically post-processing the quantum computational results. We showed that using the sample mean phase direction as a stand-alone estimator of the phase carries a lower accuracy error bound than using the standard majority rule. Moreover, it can be further post-processed to construct an unbiased estimator of the phase by inverting Eq. (9), in this initial state scenario. Despite working in the scenario where the input state is an eigenstate, the mean phase direction is also the basis for a generalization to post-processing superposition state inputs, a project we leave as future research.

We then presented the use of this newly-introduced quantity to directly find the energy levels of Hamiltonians that do not require Trotterized simulation. With this approach, we bypass the need to increase the resolution in the quantum algorithm and trade it by sampling and post-processing, avoiding the limitations associated with higher circuit depth in NISQ devices. To simulate Hamiltonians with non-commuting terms, we explored the use of the first-order Trotter-Suzuki decomposition with majority-rule QPE. The contribution of the mean phase direction for the post-processing of this scenario also requires further exploration.

On the experimental side of this project, we provided proof-of-concept demonstrations of our methods using

the quantum computing devices from IBM Q. We tested this DQS program using three simple model Hamiltonians for which the quantum circuits were provided. Two of these models were studied with the PEA and our proposed estimator, while the third one was simulated using the IPEA with the standard majority rule. To characterize dispersion of the results obtained with the non-exhaustive IPEA, we proposed a way to reconstruct a lossy probability distribution over all the possible R-bit sequences by using the two-state histograms obtained at each iteration; this approach was used to calculate the phase circular standard deviation.

Experiments showed that noise can degrade the fault-tolerant probability distributions enough to bias the estimators sometimes. Because of that, we had a second reason to execute the programs for several timesteps while varying a control parameter τ in the time propagator of the simulated systems. This allowed reducing variability by fitting the experimental data to the expected theoretical model to estimate the energy levels through the fitting parameters. As can be seen from the final results, this method proved its robustness. If necessary, the precision of the final energy level estimations of the systems implemented with non-trotterized propagators could in principle be further improved by simulating and fitting longer evolution times.

Despite demonstrating our methodology with such simple models, our methods are fully general and can be applied to more complex hamiltonians as soon as the hardware develops enough to permit so. Thanks to the exhaustive version of the IPEA, it becomes apparent that the main limitation for QPE-based digital quantum simulation on noisy near-term devices rests on the implementation the propagator of the system. Further progress requires the availability of quantum hardware with a balanced improvement of the number of qubits, connectivity, coherence time and error rates, such that it becomes possible to reliably execute wider and deeper circuits.

Appendix A: QPE with superposition states

Here we discuss the workings of the PEA [20, 21] when the initial state $|\psi\rangle$ is not an eigenstate of $\mathcal{U}$. We write

$$|\psi\rangle = \sum_n c_n |\phi_n\rangle, \tag{A1}$$

where $\sum_n |c_n|^2 = 1$, and

$$\mathcal{U}|\psi\rangle = \sum_n c_n e^{2\pi i \phi_n} |\phi_n\rangle. \tag{A2}$$

After the controlled-$\mathcal{U}$ operations, the quantum state of the computer can be written up as

$$|\Psi_{II}\rangle = \sum_n c_n \left(\bigotimes_{k=1}^{R} \frac{|0\rangle + e^{i2\pi 2^{k-1}\phi_n}|1\rangle}{\sqrt{2}}\right) \otimes |\phi_n\rangle. \tag{A3}$$

The permuted inverse quantum Fourier Transform maps the state of the phase register qubits that is associated with each $|\phi_n\rangle$ into a state of the computational basis given by

$$|\Psi_{III}^n\rangle_{\mathrm{PR}} = QFT^\dagger \left[\bigotimes_{k=1}^{R} \frac{|0\rangle + e^{i2\pi 2^{k-1}\phi_n}|1\rangle}{\sqrt{2}}\right]$$
$$= \sum_{b_{1,n}=0}^{1} \sum_{b_{2,n}=0}^{1} \cdots \sum_{b_{R,n}=0}^{1} \left(\sum_{k=1}^{2^R} \frac{e^{i2\pi(\phi_n - 0.b_{1,n}b_{2,n}\cdots b_{R,n})(k-1)}}{2^R}\right) |b_{1,n}\rangle |b_{2,n}\rangle \cdots |b_{R,n}\rangle, \tag{A4}$$

where we have expressed the phases as binary fractions

$$0.\phi_n = 0.b_{1,n}b_{2,n}\cdots b_{R,n} = \sum_{k=1}^{R} \frac{b_{k,n}}{2^k}. \tag{A5}$$

When ϕ_n can be expressed exactly in R bits, this returns a single basis state given by

$$|\Psi_{III}^n\rangle_{\mathrm{PR}} = \bigotimes_{k=1}^{R} |b_{k,n}\rangle = |0.\phi_n\rangle, \tag{A6}$$

such that we get the phase ϕ_n with probability $|c_n|^2$. When it cannot, the outcome of the PEA before the projective readout on the phase register is

$$|\Psi_{III}\rangle = \sum_n c_n |\Psi_{III}^n\rangle_{\mathrm{PR}} \otimes |\phi_n\rangle. \tag{A7}$$

Thus, after measurement, we obtain the parent distribution given by Eq. (3) with probability $|c_n|^2$. As a consequence, a given $0.\phi$ readout, e.g. 110 for the case of $R = 3$, can correspond to the collapse of different orthogonal states of the simulation register. In this way, the readout produces a distillation of the $|\phi_n\rangle$ states that is no longer a linear superposition.

Appendix B: Implementation of controlled-$\mathcal{U}$ operations in terms of elementary gates

The name of the game in phase estimation is to implement the time evolution operator $\mathcal{U}(t) = e^{-it\mathcal{H}/\hbar}$. That can be done once we have the Hamiltonian translated into the qubit language as a sum of terms $\mathcal{H} = \sum_m h_m$, each one with a tensor product structure of X, Y, Z Pauli operators. Exponentiation of each h_m term is easily carried out as follows [21].

First recall that $R_z(\theta) = \exp(-i\theta Z/2)$ acts on a general qubit state by adding a phase $\exp(-i\theta/2)$ (resp. $\exp(i\theta/2)$) to the $|0\rangle$ (resp. $|1\rangle$) basis state. Now take an operator of the form $\exp(-i\theta Z_1 Z_2 \cdots Z_n/2)$. Its effect is to apply the phase $\exp(-i\theta/2)$ (resp. $\exp(i\theta/2)$) to the computational basis states with an even (resp. odd) number of qubits in state $|1\rangle$. To reproduce this operation with a n-qubit quantum circuit, the approach lies in entangling all the qubits with a cascade of $n-1$ cX gates targeting an arbitrarily chosen *parity qubit* where a single $R_z(\theta)$ gate acts before the same cX cascade is applied again. The task of the first batch of cX gates is to ensure the parity qubit is in the $|0\rangle$ (resp. $|1\rangle$) state if parity is even (resp. odd) before the $R_z(\theta)$ gate is applied to it, while the second cascade returns this phase back to the original basis state. Two different ways of constructing the cX cascade are shown in Fig. 15. The advantage of each approach depends on the coupling architecture of the computer. Mixing both approaches is also possible.

If the exponential contains X or Y matrices, we need to change to their diagonal basis in between the full circuit, that is $X = P_X^\dagger Z P_X$ and $Y = P_Y^\dagger Z P_Y$ with $P_X = H$ and $P_Y = R_x(\pi/2)$. Thus, it suffices to apply a P_X and P_Y gate on the correct qubits before computing parity, and their adjoint at the end of the circuit, as also exemplified
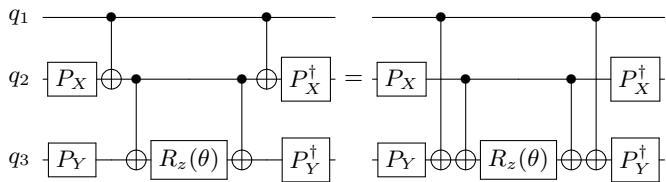
FIG. 15: Implementation of a general Hamiltonian term exemplified with $\exp\left(-i\theta Z_1 X_2 Y_3/2\right)$. Two equivalent ways of computing parity are shown.

in Fig. (15). To make it a controlled operation, only the action of the Z-rotation gate needs to be controlled, as all the other operators are applied together with their adjoints, yielding the identity when the control qubit is set to $|0\rangle$.

[1] W. Kohn, Rev. Mod. Phys. **71**, 1253 (1999).

[2] S. Langhoff, *Quantum mechanical electronic structure calculations with chemical accuracy*, vol. 13 (Springer Science & Business Media, 2012).

[3] R. P. Feynman, International journal of theoretical physics **21**, 467 (1982).

[4] D. Deutsch, Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences **400**, 97 (1985).

[5] D. S. Abrams and S. Lloyd, Phys. Rev. Lett. **79**, 2586 (1997).

[6] D. S. Abrams and S. Lloyd, Physical Review Letters **83**, 5162 (1999), quant-ph/9807070.

[7] J. Preskill, Quantum **2**, 79 (2018).

[8] I. M. Georgescu, S. Ashhab, and F. Nori, Rev. Mod. Phys. **86**, 153 (2014).

[9] A. Aspuru-Guzik, A. D. Dutoi, P. J. Love, and M. Head-Gordon, Science **309**, 1704 (2005).

[10] B. P. Lanyon, J. D. Whitfield, G. G. Gillett, M. E. Goggin, M. P. Almeida, I. Kassal, J. D. Biamonte, M. Mohseni, B. J. Powell, M. Barbieri, et al., Nature chemistry **2**, 106 (2010).

[11] P. O'Malley, R. Babbush, I. Kivlichan, J. Romero, J. McClean, R. Barends, J. Kelly, P. Roushan, A. Tranter, N. Ding, et al., Physical Review X **6**, 031007 (2016).

[12] D. Wecker, M. B. Hastings, N. Wiebe, B. K. Clark, C. Nayak, and M. Troyer, Physical Review A **92**, 062318 (2015).

[13] A. Cervera-Lierta, Quantum **2**, 114 (2018).

[14] A. Chiesa, F. Tacchino, M. Grossi, P. Santini, I. Tavernelli, D. Gerace, and S. Carretta, Nature Physics **15**, 455 (2019).

[15] J. Preskill, arXiv preprint arXiv:1811.10085 (2018).

[16] B. P. Lanyon, C. Hempel, D. Nigg, M. Müller, R. Gerritsma, F. Zähringer, P. Schindler, J. T. Barreiro, M. Rambach, G. Kirchmair, et al., Science **334**, 57 (2011).

[17] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. Obrien, Nature communications **5**, 4213 (2014).

[18] A. Kandala, A. Mezzacapo, K. Temme, M. Takita, M. Brink, J. M. Chow, and J. M. Gambetta, Nature **549**, 242 (2017).

[19] B. T. Gard, L. Zhu, G. S. Barron, N. J. Mayhall, S. E. Economou, and E. Barnes, arXiv preprint arXiv:1904.10910 (2019).

[20] R. Cleve, A. Ekert, C. Macchiavello, and M. Mosca, Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences **454**, 339 (1998).

[21] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition* (Cambridge University Press, 2011), ISBN 978-1107002173.

[22] R. B. Griffiths and C.-S. Niu, Physical Review Letters **76**, 3228 (1996).

[23] M. Dobsicek, G. Johansson, V. Shumeiko, and G. Wendin, Physical Review A **76** (2007).

[24] IBM Q, URL https://www.research.ibm.com/ibm-q/.

[25] S. J. Devitt, Physical Review A **94**, 032329 (2016).

[26] D. Alsina and J. I. Latorre, Physical Review A **94**, 012314 (2016).

[27] S. Fedortchenko, arXiv preprint arXiv:1607.02398 (2016).

[28] D. García-Martín and G. Sierra, arXiv preprint arXiv:1712.05642 (2017).

[29] J. R. Wootton, Quantum Science and Technology **2**, 015006 (2017).

[30] P. J. Coles, S. Eidenbenz, S. Pakin, A. Adedoyin, J. Ambrosiano, P. Anisimov, W. Casper, G. Chennupati, C. Coffrin, H. Djidjev, et al., arXiv preprint arXiv:1804.03719 (2018).

[31] B. Murta, G. Catarina, and J. Fernández-Rossier, Physical Review A **101**, 020302 (2020).

[32] A. Y. Kitaev, Electronic Colloquium on Computational Complexity (ECCC) **3** (1996).

[33] P. W. Shor, in *Proceedings 35th annual symposium on foundations of computer science* (Ieee, 1994), pp. 124–134.

[34] B. Higgins, D. Berry, S. Bartlett, M. Mitchell, H. Wiseman, and G. Pryde, New Journal of Physics **11**, 073023 (2009).

[35] S. Kimmel, G. H. Low, and T. J. Yoder, Physical Review A **92**, 062315 (2015).

[36] T. E. O'Brien, B. Tarasinski, and B. Terhal, New Journal of Physics (2019).

[37] K. M. Svore, M. B. Hastings, and M. Freedman, arXiv preprint arXiv:1304.0741 (2013).

[38] R. D. Somma, New Journal of Physics **21**, 123025 (2019).

[39] V. V. Shende, S. S. Bullock, and I. L. Markov, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems **25**, 1000 (2006).

[40] K. V. Mardia and P. E. Jupp, *Directional Statistics* (Wiley, 1999), ISBN 978-0471953333.

[41] B. Efron and T. Hastie, *Computer Age Statistical Inference: Algorithms, Evidence, and Data Science (Institute of Mathematical Statistics Monographs)* (Cambridge University Press, 2016).

[42] P. Jordan and E. P. Wigner, Z. Phys. **47**, 631 (1928).

[43] N. Moll, A. Fuhrer, P. Staar, and I. Tavernelli, Journal

of Physics A: Mathematical and Theoretical **49**, 295301 (2016).

[44] H. F. Trotter, Proceedings of the American Mathematical Society **10**, 545 (1959).

[45] M. Suzuki, Physics Letters A **146**, 319 (1990).

[46] M. Heyl, P. Hauke, and P. Zoller, Science advances **5**, eaau8342 (2019).